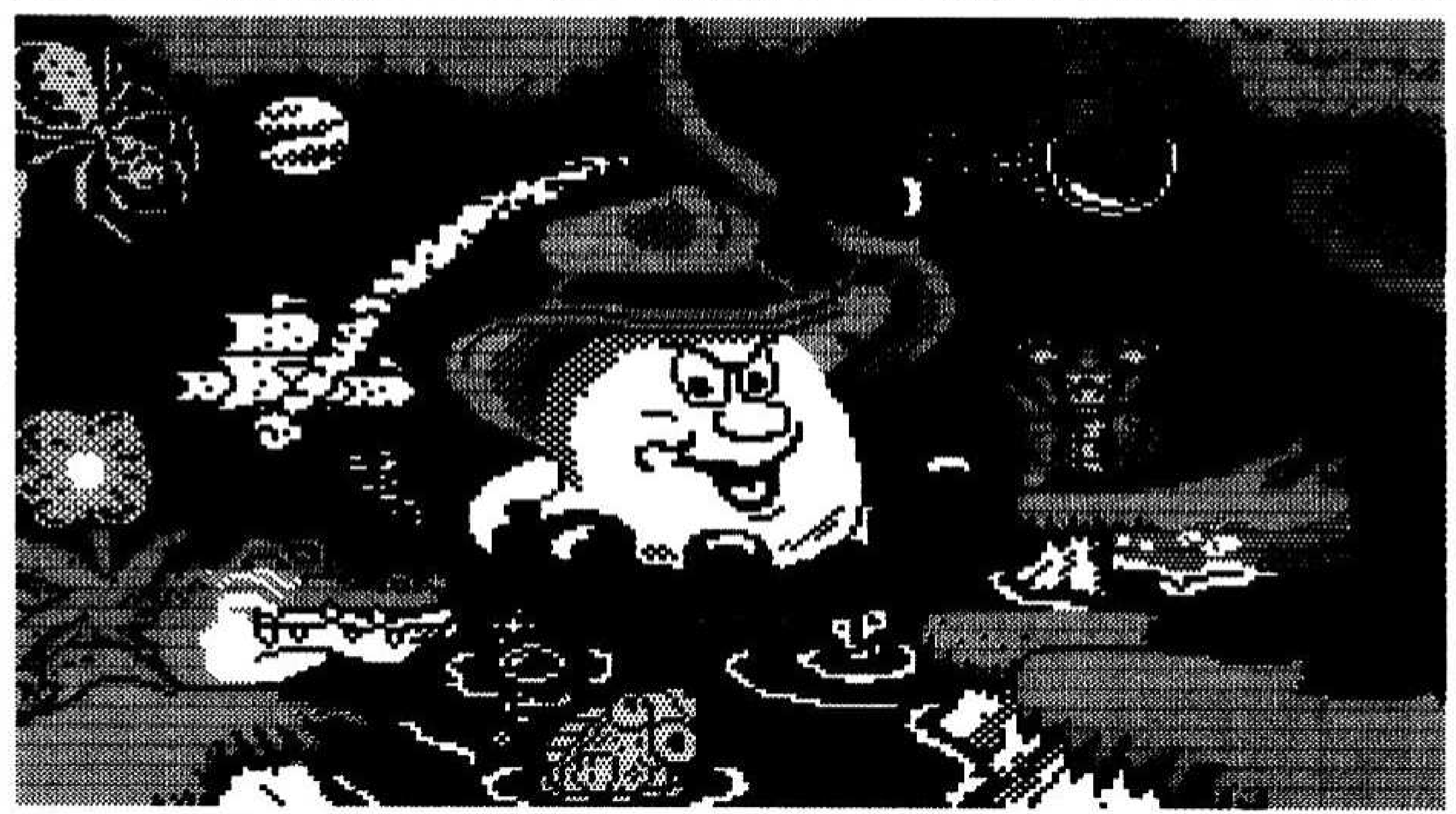
Spectrum Profi Club

für alle Spectrum und SAM Freunde



Speccy-Star Dizzy macht's vor: Mit Schwung ins Neue Jahr...

Wolfgang & Monika Haller, Tel. 0221/685946
Im Tannenforst 10, 51069 Köln
Bankverbindung: Dellbrücker Volksbank
BLZ 370 604 26, Konto-Nr. 7404 172 012

Ausgabe 97

Januar 1998

Ein frohes und gesundes Neues Jahr!

Das wünschen wir euch, dem treuen Rest von bisher 69 verbliebenen Mitgliedern im Club. Wir sind uns zwar sicher, das der ein oder andere ganz bestimmt noch folgen wird, aber realistisch gesehen nähern wir uns wirklich dem harten Kern.
Dennoch hoffen wir auch in diesem Jahr wieder Monat für Monat ein interessantes Info mit eurer Mithilfe erstellen zu können. Diesmal bitten wir euch auch, euer Augenmerk auf den neuen Wettbewerb zu richten, den wirklich viele mitmachen können und dies hoffentlich auch tun. Doch nun geben wir erst einmal weiter an Hans Schmidt, der sich so seine ganz eigenen Gedanken gemacht hat.

Einige Gedanken und das Tastatur-IF

Hallo, ihr Spectrumfreunde!

Wieder ist ein Jahr vergangen. Jeder wird hoffen, daß das kommende Jahr wenigstens so wird, wie es das vergangene war, wenn möglich, sollte es besser werden. Auf unser gemeinsames Hobby bezogen, hatte wohl ein jeder so seine eigenen Beziehungen und Erlebnisse. Hard- und softwaremäßig gab es noch immer Bewegung und das ist auch gut so, denn sonst gäbe es Stillstand und das wäre ja das Aus. Aber soweit ist es noch lange nicht.
Trotzdem macht man sich so seine Gedanken. Wie kommt es bloß dazu, das man sich für einen Tag ins Auto setzt, über 600 Kilometer fährt, nur um für ein paar Stunden mit Gleichgesinnten und Freunden zusammen zu sein? Da muß doch irgendeine treibende Kraft dahinterstecken. Nun, ich gebe zu, daß es wohl nicht nur uns so gehen wird. Es gibt viele Menschen, die ihrem Hobby genauso frönen. Doch oft ist es ja auch so, daß man nicht nur mit den Freunden reden will, man hat ja auch was vorzuweisen, war fleißig im stillen Kämmerlein, hat gebastelt und programmiert. Und gerade das erfüllt einen dann mit Freude und Stolz. So setzt man sich mit neuem Elan in sein Auto, fährt wieder 600 Kilometer zurück.
Gern denke ich noch heute an das letzte Treffen in Mönchengladbach. Man kennt sich schon, ist nicht mehr so fremd. Gerade auch der Erfahrungs- und Gedankenaustausch mit den holländischen Usern ist für mich immer ein Höhepunkt des Treffens.
Aber, und das habe ich von Anfang an gespürt, es gibt keine Vorurteile untereinander. Jeder gönnt jedem seinen Erfolg, jeder gibt jedem, ohne sich zu zieren, von seinen geistigen oder materiellen Dingen etwas ab. Da spielt es keine Rolle, ob jemand alt oder jung ist, ob er aus Ost oder West kommt. Und jedesmal knüpft man neue Bekanntschaften.
Nun hatte ich ja mit meinem Tastaurinterface so meine Schwierigkeiten. Im Info 5/97 hatte ich ja darüber ausführlich berichtet. Aber auch noch in Mönchengladbach zum Treffen konnte ich den Effekt vorführen, daß die Tastenkombination SS + A, also u.a. der Sprung zum Menu in Tasword II, nicht ging. Ebenso der Stopbefehl und die Kombination CS + s, also das große S.
Jean Austermühle (Hallo!!) tippte auf die Folie. Als er mich in Berlin besuchte, brachte er mir eine mit. Nach einer Woche probierte ich sie aus, aber es blieb der gleiche Effekt.
Da ich in meinem 128er noch zwei weitere Betriebssysteme habe, das normale 82er und das 87er ISO-ROM, konnte ich wählen. Im 87er ging der Stopbefehl in Basic, aber auch nicht im Tasword. Da kam ich ganz schön ans grübeln. Der erste Gedanke war, das EPROM hat eine Macke. Aber alle Testereien brachten kein sinnvolles Ergebnis.
Ein Lichtblick brachte das Ablöten der entsprechenden Leitungen an der Tastatur. Wenn ich diese direkt verband, funktionierte es. Aber da ich keine Tasten außerhalb der Tastatur haben wollte, mußte ich weitersuchen. Der Gedanke kam, daß es am Kabel liegen könnte. Obwohl da keine HF anlag, sollte es ein Übersprechen von einer Leitung zur anderen geben?
Der nächste Versuch brachte dann die Lösung: Es liegt an der Folie. Der Widerstand der Leitungen ist mit 40 bis 60 Ohm für das IF zu groß. Ein Glück, das ich "damals" bei A bis Z mir zwei Tastaturen mit mechanischen Kontakten gekauft hatte. Jetzt holte ich sie hervor und machte mit den wichtigsten Adress- und Datenleitungen einen Test. Alles o.k.!!
Also, die ganze Tastaturplatte, wie im Info 11/94 schon mal beschrieben, verdrahtet. Jetzt kann ich wieder richtig mit meinem Spectrum arbeiten. Die Tasten Y und Z vertauscht und die Sonderzeichen mit der Elektronik verbunden - und die deutsche "Profitastatur" ist fertig.
Wolfgang (Wo) riet mir, ein Interface für eine PC-Tastatur zu kaufen. Aber da war mir der Preis doch etwas zu happig. Runde 100 DM für ein paar IC! Dazu noch die Tastatur für auch noch 50 DM. Nein, da bastelte ich lieber und hatte den Erfolg selber!
So, liebe Spectrumfreunde, das war noch als Ergänzung zum Tastatur-IF gedacht. Traut euch nur ran, ihr habt dann nur noch Freude mit dem Rechner!!!
Noch ein Nachtrag an Jean: Das 128er Beta IF geht nicht! Kein Zeichen, keine Meldung auf dem Bildschirm. Das 3,5" Laufwerk sagt auch nichts, wenn ich ihm Strom gebe läuft zwar der Motor, aber er hält nicht an, wie es bei den 5,25" Laufwerken der Fall ist.

Hans Schmidt, Fredersdorfer Straße 10
10243 Berlin, Tel. 030/296 1959

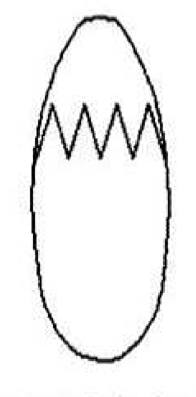
Das Jahr ist rum - der Pokal auf meinem Regal wäre schon eingestaubt, wenn ich zwischenzeitlich nicht umgezogen wäre, und eine Idee geistert mir schon seit Monaten im Kopf herum - Zeit für den nächsten

Wettbewerb

Nein, keine Mathematik, die das Hirn zum Qualmen bringt, keine Programmierkniffe sind gefragt!
Diesmal soll's was für alle sein, die mal wieder Lust haben, ein paar Tasten eines SINCLAIR- oder SAM-Rechners niederzudrücken, um 3 oder 4 verschiedene Befehle zu einem Programm zu komponieren! Alles was ihr braucht, sind die allseits beliebten Grafikbefehle PLOT und DRAW sowie der Befehl, mit dem ihr vermutlich in grauer Vorzeit eure ersten Schritte in BASIC gewagt habt: PRINT.
Der Haken an der Sache? Kreativ müßt ihr sein!

AUFGABE

Programmiert eine originelle Strichzeichnung - zeichnet mit einfachen Befehlen ein Tier und gebt dem ganzen einen Titel! Als Vorgabe erhaltet ihr von mir einen Zacken, den ihr in das Bild integrieren sollt. Nebenstehend seht ihr, was mir dazu eingefallen ist.
Das ganze soll ausschließlich mit Grafikbefehlen auf weissem Hintergrund gezeichnet werden, also ohne UDGs, PRINTs oder fertige Screenteile.



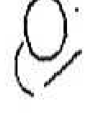
Styracosaurus beim Ausschlüpfen

ALLER ANFANG...

Keine Idee? Fragt Frau, Freundin, Partner,... Setzt euch zusammen mit den Kindern vor den Compi, die werden euch schon sagen: "Mach da noch'n Strich hin!" - "Du mußt dem noch einen Schwanz malen!"
Und dann geht probieren über studieren.
Da WoMo und ich möchten, daß viele von euch mitmachen, gibt es eine Belohnung für den 10. Einsender, siehe weiter unten.

WIE'S GEHT?

Jongliert mit den Grafikbefehlen PLOT und DRAW; auch CIRCLE, wenn ihr wollt. Mit DRAW x,y könnt ihr Linien zeichnen, mit DRAW x,y,z Bögen. Für z am sinnvollsten sind Werte zwischen -PI und PI, probiert's aus!
SAM-User benutzen Screen-MODE 1 und kein FILL, um gleiche Chancen zu gewährleisten.



Für den Zacken müßt ihr diese Programmzeile einbauen:

```
xxxx DRAW 10,20: DRAW 10,-20
```

Zu viel Striche, Punkte usw. dürfen es aber nicht werden. Die Anzahl ist begrenzt, da ihr insgesamt maximal 60 Zahlen im Programm benutzen dürft! Inclusive dem Zacken sowie ALLEN in Befehlen verwendeten ausgeschriebenen Zahlen, z.B. auch in FOR, LET etc. Ich möchte damit Workaholics von der Sisyphusarbeit abhalten, ein Bild aus einzelnen Pixeln zu konstruieren...

Falls jemand in einer anderen Sprache als BASIC schreibt, so soll er die Routinen benutzen, welche dieselben Ergebnisse wie die erwähnten Befehle liefern.

Genau ein PRINT soll schließlich den Titel des Bildes, z.B. den Namen des Tieres, auf den Bildschirm bringen.
Ansonsten laßt eurer Programmierlust freien Lauf.

Hier mein Listing zu obigem Bild:

```
  1 REM Wettbewerb Tierzeichnung
       blödes Beispiel von Ingo
 10 PAPER 7: CLS
 30 PLOT 80,70
 40 DRAW 40,-40,1.5
 50 DRAW 40,38,1.5
 60 DRAW -20,82,1
 70 DRAW -20,10,1
 80 DRAW -20,-10,1
 90 DRAW -20,-80,1
100 PLOT 80,107
110 FOR i=1 TO 4
120  DRAW 10,20: DRAW 10,-20
130 NEXT i
200 PRINT AT 20,0;"Styracosaurus beim
    Ausschlüpfen"
```

JURY

So ich alle eure Bilder beisammen habe, werde ich eine Jury bitten, jedes Bild danach zu bewerten, ob ein Tierchen erkennbar und wie originell es dargestellt ist.
Die Jury wird ebenso viele Mitglieder haben wie der Wettbewerb Teilnehmer. Sie wird weiblich und männlich besetzt und altersmäßig breit gestreut sein.

WAS WANN ABSENDEN?

Schickt mir am besten eine SAM- oder +D-Diskette, Opus und Kassette tun's auch. Drauf muß natürlich das Programm sein, bei anderen Sprachen als BASIC der kompilierte Code sowie der Quelltext als Listing. Bei besonders kurzen BASIC-Programmen reicht notfalls ein Listing.
Einsendeschluß ist der 23. Februar, so daß ich im Märzheft den Gewinner küren kann.

ANREIZ

Dem 10. Einsender wird ein besonderer Spaß zu Teil: Sein Tier erscheint auf dem Titelblatt des folgenden SPC-Heftes! Falls es weniger Einsender geben sollte, so wird der 2. damit belohnt.
Dann gibt es natürlich noch den von Peter Meindl gestifteten Wanderpokal für den Sieger und vielleicht auch noch ein Bonbon, WoMo (Klaro Ingo, wir sind dabei! Vielleicht sollten die Einsender zwei oder drei (erfüllbare) Hard- oder Software-Wünsche nach dem verwendeten Computertyp notieren, auf die wir dann zurückgreifen können.)?

Ich wünsche allen Teilnehmern viel Vergnügen und ein gesundes Augenmaß beim Abschätzen der Zahlen!

Ingo Wesenack, Dahlmannstr. 10, 10629 Berlin
Tel: 030/3245707, e-mail: ingw@cs.tu-berlin.de

BDOS Version 1.2 Information

1997/1998 by Edwin Blink.

Tolle Nachricht für uns SAM-Freunde zum Jahresanfang von Edwin Blink. Er sandte uns die BDOS-Version, die für den Betrieb des von Persona vertriebenen Harddisk-Interfaces (wir schrieben schon mehrfach darüber) nötig ist samt nachfolgendem File (in englisch), welches zeigt, wozu BDOS fähig ist. Die BDOS-Version ist Shareware, funktioniert auch ohne Harddisk und kann von uns weitergegeben werden.
Wer noch keine Harddisk hat, sollte es sich nun wirklich überlegen. Der Vorteil bei dieser neuen Version ist z.B., das die Harddisk wie ein 2. Laufwerk arbeitet. Damit entfällt die für die Urversion fast unentbehrliche 1 MB Speichererweiterung, die auch kaum zu bekommen ist. Obwohl mein System zu meiner vollen Zufriedenheit arbeitet (aber nur dank der genialen Utilities von Ian Spencer), darf man nicht verkennen, daß das HDOS offensichtlich nicht vorankommt (und eventuell mal wie Beethovens "Unvollendete" in die Geschichte eingehen könnte).
Doch nun zurück an Edwin Blink.

Introduction
B-DOS Is an improved SAMDOS. B-DOS was mainly written to use a harddisk on the SAM using the internal harddisk interface. However B-DOS can also be used without a harddisk but it only works with one diskdrive.
To make using a harddisk as user friendly as possible, the hard disk works like a virtual second disk drive. The storage capacity of a harddisk is devided into 800K sections which act like a floppydisk. (10 sectors per track, 80 tracks and 2 sides). Such a section is called a RECORD. A record is selected by the same basic command.
B-DOS has beside the all SAMDOS features a lot of new features.

BOOTING
When B-DOS is booted (BOOT and no DOS in memory or BOOT 1): The mode 3 screen is selected and the B-DOS version is displayed. B-DOS then starts looking for a hard disk and if found, it will print the type of harddisk connected and the harddisks specifications. It will also print how many records are available on the harddisk.
The DOS variables are automatically set to use the harddisk so no manual settings are required.
If a harddisk is connected but is not recognized, try press 'R' to retry.

BDOS BASIC COMMANDS
Note: All between brackets is optional and should be entered without the brackets.
Example: DIR 1 (!) Enter: DIR 1 or: DIR 1!

Hard disk motor control
DEVICE ON: Puts the harddisk in idle mode (Motor on).
DEVICE ON,n: As above and sets the APD counter (Auto Power Down). The harddisk enters stand-by mode when the time set by the APD counter has elapsed.
n=0 disable APD counter
n=12/240 APD active after n • 5 seconds.
Note. The APD commands may work different or not work at all with certain harddisks.
DEVICE OFF: Puts the hard disk in stand-by mode (Motor off) Idle mode is entered as soon as a new command is issued.
DEVICE OFF,n: As above and sets the APD counter.
DEVICE STOP: Puts hard disk in sleep mode (motor off and hard disk is no longer accessable. Also DEVICE D2 is set to non exist and DEVICE is set to D1. Use DEVICE ON to re-activate.
RESTORE DEVICE: Resets hard disk logic and selects DEVICE D2

RECORDs
The harddisk capacity is devided into 800K sections which are called records. A record must be selected before it can be used. A record can be selected by:
RECORD n: Selects record number 'n' as permanent record
RECORD "name": Selects record named "name" as permanet record. Before a record is selected and DEVICE D2 is set a test is made if that record is available to B-DOS. (Records can be reserved for other uses). The 'Invalid record' report will be given if a record is not available to B-DOS. (See FORMAT for more info).
RECORD can be used in combination with outer commands. in that case a record is only selected to be used by that command and can be in three forms: RECORD, RECORD n or RECORD "name". To make things easier the RECORD keyword can be entered at once pressing SYMBOL Z. Records and disks can also be named with a name up to 16 characters (See RENAME).
The following command can be used to list all records with a name:
RECORD: Will list all named records with their record number. Record names which start with a

CHR$ 0 (no name) will not be displayed.
RECORD (#s): Same as above but output to stream s
RECORD (#s),"*": Same as above but using wild-cards

Directory
DIR only or DIR added with '!' at the end will perform a short directory (filenames printed only). In all other cases a detailed directory will be given.

DIR drive (!)	DIR #stream,drive (!)
DIR drive,"*" (!)	DIR #stream,RECORD (!)
DIR RECORD (!)	DIR #stream,drive,"*" (!)
DIR RECORD ,"*" (!)	DIR #stream,RECORD,"*" (!)

RECORD may be: 'RECORD' only, 'RECORD n' or 'RECORD "name"'.
The file number of a file is printed in inverse if that file is protected. In screen mode 3 the execute address of a CODE file is also printed (third number). If BIT 0 of DVAR 38 and screen mode 3 Date and time will be printed if the file has a valid date.

Erasing files
ERASE "filename" (?): Erases normal file
ERASE OVER "filename" (?): Erases a protected file
(?) will activate 'Erase "" (Y/N/C/E) ?' request for each file to copy press Y to copy the file, N to skip the file,C to continue copying without the request, E to end the command. Any other key will be seen as 'N'.

Protecting files/records
PROTECT "filename": Protects a file
PROTECT OFF "filename": Unprotects a file
PROTECT RECORD (backslash) n (backslash) "name": Software write protect to protect a complete record against any write operation (FORMAT too)
PROTECT OFF RECORD (backslash) n (backslash) "name": Removes record write protect

Hiding files
HIDE "filename": "filename" will not be displayed in a DIR.
HIDE OFF "filename": To undo a HIDE

LOAD n: LOAD a file by number. As given in a detailed DIR. Note: The file is loaded from the last used drive.

Sector commands
READ AT d,t,s,a,n: Reads 'n' sectors from drive 'd' starting at track 't', sector 's' to ram at 'a'. Note: 'n' must be in the range of 16384 to 540671. All sector formats (128/256/512 or 1024) can be read. However multiple sectors will only work correctly with standard disk format (10 sectors per track numbered 1 to 10).
WRITE AT d,t,s,a,n: Writes sectors to disk. See READ AT also.
VERIFY AT d,t,s,a,n: Checks if contents of sectors is the same as in ram. See READ AT also.

Renaming files/records
RENAME "oldname" to "newname": Renames "oldname" to "newname". Note: The new name can be the same as the old name to allow case changes.
RENAME TO "Label": Renames a disk/record to "Label"
RENAME RECORD TO "label": Renames record to "Label"
(RECORD can be: RECORD, RECORD n or RECORD "name")
Use CHR$ 0 as "label" to remove a disk/record name .Note: There is no check if records have the same name.

FORMATTING DISKS/RECORDS
FORMAT (OVER) (RECORD) ("name"):
OVER Formats immediately without a request.
RECORD may be:
RECORD: Formats last selected record
RECORD n: Formats record number n
RECORD "name" Formats record named "name".
"name" names disk/record with name "name". If 1st char of name is a CHR$ 0 or no name is given, the disk/record will not be named.
Before a disk is formatted a check is made if the disk has been formatted before and if so 'FORMAT "" (Y/N)' is given.
B-DOS formats both sides at the same track at once (SAMDOS/MASTERDOS formats all tracks on side first and then side two), which makes a format about 20% faster.

ote 1: Harddisks are hard sectored and can not be formatted. B-DOS will fill all 1600 sectors in a record with zeros instead.
Note 2: B-DOS sets an BDOS ID in the first directory entry. Before a record is selected a check is made for this ID. If the ID is not there the record is not selected. This allows records to be reserved for non B-DOS use. To select a record without a ID you can DPOKE the record number directly in DVAR 25. and do a POKE DVAR 27,0: to disable a possible software write protect from the last selected record.
To set the ID in that record you can enter FORMAT OVER or RENAME TO CHR$ 0 (or any other name you like). When using a harddisk for the first time, load the file 'formatter' to format the entire hard disk.

COPYING FILES
COPY (OVER) source TO target (?): (OVER) Will inhibit the 'OVERWRITE "" (Y/N)' request if a file on the target disk/record has the same name as the file to copy.

(?) will activate 'COPY "" (Y/N/C/E) ?' request for each file. To copy press Y to copy the file, N to skip the file, C to continue copying without the request, E to end the command. Any other key will be seen as 'N'. Source and target may be any of the following:
"*": All files using the current device.
"d1:*": All files using drive 1
"d2:*": All files using current harddisk record
RECORD: As above
RECORD,"*": As above
RECORD 1: As above but using record 1
RECORD 1,"*": As above
RECORD "name": As above but using record named "name"
RECORD "name","*": As above. "*" may be replaced with a string to copy selected files.

ZX FILES and SNAPSHOTS

BDOS 1.2 will copy ZX files including SNAPSHOTS correctly. Also BDOS prevents ZX files to be in the first directory entry (the one starting with number one in a detailed directory). The first directory entry holds part of the disk/record name at the same place as ZX file info.(As with Masterdos. However, Masterdos allowes ZX files to be in the first directory entry). A ZX file with corrupted file info will be corrected by BDOS when copied.
SNAPSHOTS: loading 48K SNAPSHOTS works as with SAMDOS/MASTERDOS. Before loading them the emulator files 'skelt.bin' and 'snapt.bin' must be loaded in memory before loading a snapshot. Use LOAD n to load a SNAPSHOT. When the break button is pressed while A SNAPSHOT is running, the border will change and waiting to press one of the following keys to be pressed:
'2' return to SNAPSHOT
'3' Save screen to D1 or to D2 if 'SHIFT' pressed
'4' Save SNAPSHOT to D1 or to D2 if 'SHIFT' pressed
'X' Return to SAM basic
Enter CALL MODE 1 in basic to return to the SNAPSHOT again.

DVARS (DOS VARIABLES)

0	1	Border mask 0=no border change, 1-7 border changed
1	1	Drive 1 data (normaly 208 for 2*80 tracks)
2	1	Drive 2 data (208 if harddisk connected may be less if last record is selected)
3	1	Default device chr. Set when BDOS has been booted
4	1	Default device no. Set when BDOS has been booted
5	1	Space replacement character (DIR).
6	1	Drive data last record available.
7	1	Version number /10 -10 (version 1.1 = 1)
8	1	Harddisk base sector of current record
9	1	Harddisk number of sectors per track + 1
10	1	Harddisk base head of current record
11	1	Harddisk number of heads
12	2	Harddisk base cylinder of current record
14	2	Harddisk number of cylinders
16	2	Harddisk heads * sectors
18	2	Total number of sectors on harddisk MOD 65536
20	1	Total number of sectors on harddisk/65536
21	2	Number of reserved sectors on harddisk for BOOT sector and RECORD list. Equal to INT ((records+63)/32))
23	2	Total number of records available
25	2	current record selected. May be DPOKEd manually to select a record. The write protect status of a record is not updated if this DVAR is DPOKEd.
27	1	Write protect status of current record. Note: only updated if a record is selected using the RECORD command.
28	2	Temporarely record number
30	2	Alternative record number
32	2	Reserved
34	1	Harddisk select (&A0 Master/&B0 slave)
35	3	Call this dvar to re initialize Master/slave harddisk RECORD 1: POKE DVAR 34,&B0: CALL DVAR 35 to use a slave hard disk.
38	1	Date flag bit 0 = 1 Prints dates on a detailed directory in screen mode 3
39	1	Day
40	1	Month
41	1	Year
42	1	Hour
43	1	Minutes

DVARS 39/43 are used to date stamp a file if day <> 0. All values are decimal.

Edwin Blink, Kremersheerd 63
NL-9737 PK Groningen, Holland

Simcoupe im Internet

Für alle, die sich die aktuelle Version des **Simcoupe** aus dem Internet ziehen wollen, hier die Adresse (gefunden von Scott-Falk Hühn):
http://www.hep.ucl.ac.uk/~ajs/simcoupe/simcoupe_dos.html

BASICODE

- the never ending story...

Jeder der mich kennt, weiß, daß ich recht beharrlich an stehenden Problemen arbeite. Auf direktem Weg bin ich ja über BASICODE 2, wegen fehlender ZX Saveroutine, nicht zum Ziel gekommen. Deshalb habe ich mir einen KC85/4 (Rechner der ehemaligen DDR) besorgt, um vor Ort, beginnend mit BASICODE 3 (für KC), meine Versuche zu starten. Ein weiterer Grund des Beschäftigens mit dem KC waren die Datenverwaltungs-Programme von Prof. HENRION eingeben und mit Originalgrafik laufen lassen zu können.
Kurz zur Vorstellung des KC85/4 einige technische Details:
Prozessortyp: U 880 D; Schreib-/Lesespeicher: 128 KByte dRAM, für Anwender nutzbar ca. 64 KByte; Festwertspeicher: 20 KByte ROM; Bildaufbau: vollgrafisch, 320 x 246 Bildpunkte; Farben: 16 Vorder- und 8 Hintergrundfarben; Tonerzeugung: 2 Tongeneratoren; Tonhöhenumfang: 2 x 5 Oktaven; Anzahl der Tasten: 64; Frei programmierbare Tasten: 6, doppelt belegbar; Programmiersprachen: BASIC (eingebaut), FORTH (Modul oder Software), PASCAL (Modul oder Software), LOGO (Software), Assembler (U880)
Es existiert ein Diskettenaufsatz D004, der aber nur noch aus 2. Hand erhältlich ist.
Für mich verblüffend war, daß ich ein BASICODE 3-Programm, welches mir Wolfgang Haller am Spectrum abgesaved hatte, gut in den KC85/4 einladen konnte, selbst Teile der HRG-Graphik liefen gut.
Von KC-Usern habe ich erfahren, daß es auch ein BASICODE 2 für den KC gibt. Desweiteren habe ich nochmals Verbindung zu Herrn Dr. Raats in Holland aufgenommen, um den BASICODE 2 Saver für den ZX 81 zu erhalten. Mal sehen, obs diesmal klappt.
Erstaunlich für mich war allerdings auch, daß auch die Originalprogramme in BASICODE 3 in holländisch so problemlos einladbar waren, obwohl sie doch einige Jahre auf dem Buckel haben. Vorteilhaft ist die Kontrollfunktion bei der Programmeinlesung.
Die bisherigen Versuche lassen die gegenseitige Softwarenutzung der Sinclair-Computer mit anderen, z.B. auch KC und umgekehrt unter der genialen BASICODE-Idee als praktikabel erscheinen.
Auf diesem Gebiet werde ich weiter probieren, mein Hauptinteresse gilt dabei naturwissenschaftlichen Programmen, aber auch guten Anwenderprogrammen (Datenbanken, Textverarbeitung), natürlich aber auch guten HRG-Spielen zur Auflockerung.

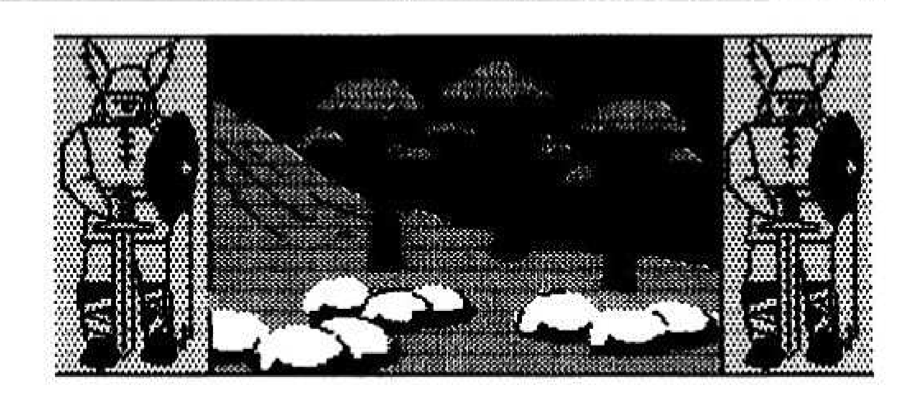
**Henning Räder, Emmericher Straße 35
46147 Oberhausen, Tel. 0208/688969**

Was ist eigentlich BASICODE?

Ich habe mir für dieses Jahr vorgenommen, zu einigen Artikeln, die nicht unbedingt allgemeine Sinclair-Themen behandeln, soweit es mir möglich ist, etwas Erklärendes beizutragen.
Basicode ist einer der allerersten Versuche, eine Programmiersprache zu entwickeln, die von möglichst vielen 8-Bit Rechnern verstanden wurde. Im Prinzip handelt es sich um ein Programm, welches erlaubt, eigene Programme in einer Art Basic zu schreiben und in einer auch für andere Rechner lesbare Form abzuspeichern (sofern diese ebenfalls ein Basicode Programm benutzen). Umgekehrt erlaubt es das Laden von eben solchen Programmen, z.B. auf unserem Spectrum, ganz gleich ob diese auf einem ZX 81, Atari XT, C 64, KC-Rechnern o.a. geschrieben wurden.
Basicode wirkt also einerseits als Übersetzer und andererseits als Programmiersprache.
Grundlagen zum Programmieren in Basicode sind: die Programmiersprache Basic, der Basic-Dialekt des Computers, die unter Basicode erlaubten Anweisungen, Funktionen, Operatoren und Variablennamen, die Basicode-Subroutinen und gewisse Regeln im Programmaufbau.
So darf z.B. ein Programm unter Basicode erst ab Zeile 1000 anfangen, da die Zeilen bis 950 mit computerspezifischen Basicode-Subroutinen belegt sind. Dagegen ist der Spectrum aber nicht auf Zeilennummer 9999 limitiert, sondern kann nun bis Zeile 32767 gehen.
Eine weitere Besonderheit ist, das man auf dem Spectrum nun mit einer Zeilenbreite von 40 Zeichen arbeiten kann (bei manchen Rechnern sogar 60 Zeichen).
Auch beim Gebrauch von Variablen wird der Spectrum User mit einer Neuigkeit überrascht: Variablen dürfen nur noch zwei Zeichen lang sein, z.B. X1 oder XY. Nicht erlaubt sind hingegen Variablen, die mit dem Buchstaben 'O' beginnen. Außerdem müssen einige spezifische Variablen benutzt und vor Programmstart mit einem Wert belegt werden, z.B. HO und VE für die Cursor- oder Textpositionierung auf dem Bildschirm oder IN für die Tastaturabfrage.
Dafür wird man andererseits entschädigt, da nun auch zwei Zeichen lange Stringvariablen erlaubt sind, wobei es auch hier einige Vorgaben gibt, die definiert werden müssen.
In der Glanzzeit des Basicode gab es sogar einen Sender in Holland, der regelmäßig Basicode-Programme über den Äther sandte.
Es gäbe noch etliches zum Programmieren unter Basicode zu sagen, aber warum schaut ihr es euch nicht mal selber an? Denn Basicode (Version 3C, C für Farbe) ist PD und von uns samt über 100 Programmen dazu zu erhalten und kann leicht an jedes Diskettensystem angepaßt werden.

Wo vom WoMo-Team

The Saga of Erik the Viking



Hallo Adventurefreunde!!

Bei unseren Abenteuerreisen durch die Welt der Spectrum Adventures hat es uns heute ganz in den Norden verschlagen, nach Schweden zu den Wikingern. Damit dürfte Freunden dieses Spielgenres bekannt sein, womit wir uns heute beschäftigen wollen. Es geht um Erik the Viking. Bei diesem Programm aus dem Hause Level 9 Computing schlüpfen wir in die Gestalt eben dieses Erik und haben dabei einige Rätsel zu lösen. Wie immer in Programmen aus dem Hause Level 9 ist es nicht ganz einfach der Lösung nahezukommen. Ganz abgesehen davon, daß wir in diesem Adventure sehr viele Seereisen unternehmen müssen. An dieser Stelle gleich mal ein Wort zu den Seereisen. Wir haben auf dem beiliegenden Plan die Reiserouten nicht erfaßt, da es manchmal nötig ist eine Richtung öfters anzusteuern. Beim nächsten Spieldurchlauf ist es wieder anders. Dies würde unserer unmaßgeblichen Meinung nach nur Verwirrung stiften. Wir raten euch deshalb, die von uns vorgeschlagenen Reiserouten einzuhalten, da sie meist ans Ziel führen. Ansonsten ist es ratsam die Bildschirmtexte auf den Seereisen aufmerksam zu lesen, da sie oft recht hilfreich sind um ein bestimmtes Ziel zu erreichen. Die einzige Location die die Seereise betrifft ist in unserem Plan also die Nummer 18. Wenn wir ein Ziel erreicht haben, so beginnt unser Plan immer an Bord des Schiffes unmittelbar vor dem Landgang. Soviel dazu. Kommen wir zu den Locations, die wir auf unserer Reise antreffen:

001) On the mountainside
002) On a path
003) On a cart-track
004) In a barn / kindling, whetstone
005) On a farmyard A
006) On a farmyard B
007) On a farmyard C
008) In the smithy / hammer, nails, scales
009) In the kitchen / black stewpot of stew
010) In the great hall
011) In the middle of a platform / trestle tables, assorted rushes, planks, blueblade
012) In furs on a bench / bedding, horn
013) On a grassy hill
014) On the shore A
015) In the boathouse / Ship Golden Dragon
016) On board of the ship
017) On the hull under the deck / skates, medallion
018) Golden Dragon is on the sea
019) On deck on the shore
020) On the shore near the deep fjord/ driftwood
021) In the forest A
022) In the forest B
023) In dense forest
024) On a ledge A
025) On a rock platform
026) In a dark cave
027) In a red room A / challice, mirror
028) On the ship near an iceberg ledge
029) On an iceberg ledge
030) In an ice cave
031) In a cold tunnel
032) In a frost cavern
033) In an ice passage
034) In a giant bedroom
035) In a cold cellar / plug
036) On the sea at the immense dragon
037) On the mast
038) On the dragon's nose
039) In a huge nostril
040) In a twisty maze / feather
041) On a rock in the Maelstrom
042) On the ship near a rocky shore
043) On a rocky shore
044) On the clifftop
045) In a passage
046) In the enchanter's hall / tapestry
047) In a yellow room A
048) In a red room B
049) In a red room C
050) In a yellow room B
051) In a red room D
052) In a yellow room C
053) In the enchanter's study / cupboard, lever, dust, scroll, green bean, spectacles, tube, rag bag

054] In the ground A
055] On the ship near a stone quay
056] On a stone quay / spellhound
057] On a golden road / spellhound
058] On marble slabs / spellhound
059] In the long gallery / spellhound
060] In the throne room / spellhound, oak chest, flask, jar, spittoon, yellow ribbon
061] On the ship near slimy rocks
062] On slimy rocks
063] In the byre of a croft / haystack, needle
064] In a stone room / tools, clippers, spade
065] On the ship near a gravel beach / helmet
066] On the gravel beach
067] On steep rocks
068] On a cliff ledge
069] On a ledge B / stick, silver bracelet, monument, untidy nest, eagle
070] On the ship near a loch
071] On the ship on the shore
072] On the shore B
073] On a grassy bank / silver amulet, pool
074] On the ship near a shady cove
075] On the shady cove
076] On the right track
077] On the path to a stave church / tombstone
078] In the church
079] In the chancel / bell, candle, book
080] On the ship near Jorvik Wharf
081] On Jorvik Wharf
082] On a junction
083] On a street
084] On the garden path
085] In a house / tabby cat
086] On the ship near a dolphin
087] Swimming in the sea
088] On the ship near a sheltered beach
089] On a sheltered beach
090] On a hill path
091] On the hill / granite slab
092] In the cairn
093] In the ground B
094] In a tight tunnel
095] In a sloping tunnel
096] On beaten ground
097] In a hot passage
098] In a round cave / seven snoring dwarfs
099] On the ship near a sandy shore
100] On the sandy shore
101] On a fertile valley / stone tree, stone fruit, mountain roots
102] On ship near the farthest shore
103] On the farthest shore
104] On rank grass
105] On a bleak plain / dogfighters
106] On crushed rock
107] On wasteland
108] On jumbled rock outside a prison/ inscription, wolf
109] At Erik's family

Soviel zur Umgebung. Kommen wir nun zu der schrittweisen Lösung dieses Adventures. Wir stehen an unserem Ausgangspunkt - On the mountainside - und starten folgendermassen:

D, W (wir erfahren, daß Erik's Familie entführt wurde), S, get kindling, exam kindling (braucht man zum Feuermachen), get whetstone, exam whetstone (auch zum Feuermachen), N, W, W, N, N, get hammer, S, W, S, E, D, exam bedding, get horn, U, exam rushes, get blueblade, exam blueblade, exam table, break table, look, get planks, W, N, E, N, N, W (on the shore), alles ablegen, inv, E, S, S, W, get stewpot, E, N, get nails, get scales, exam scales (dient dazu, Silber zu wiegen), N, W, get hammer, get horn, get planks, NE, repair ship, blow horn (Erik's Freunde - ziemlich finstere Gesellen - kommen herbei), drop horn, pull ship, get kindling, get whetstone, get blueblade, in, alles ablegen, inv, D, get skates, get medallion, exam medallion, U, alles ablegen, W, W, N, W (wir sollten jetzt an der Küste (Shore) sein - wenn nicht weiter rumsegeln bis wir was von einem Fjord lesen - dort müssen wir hin).

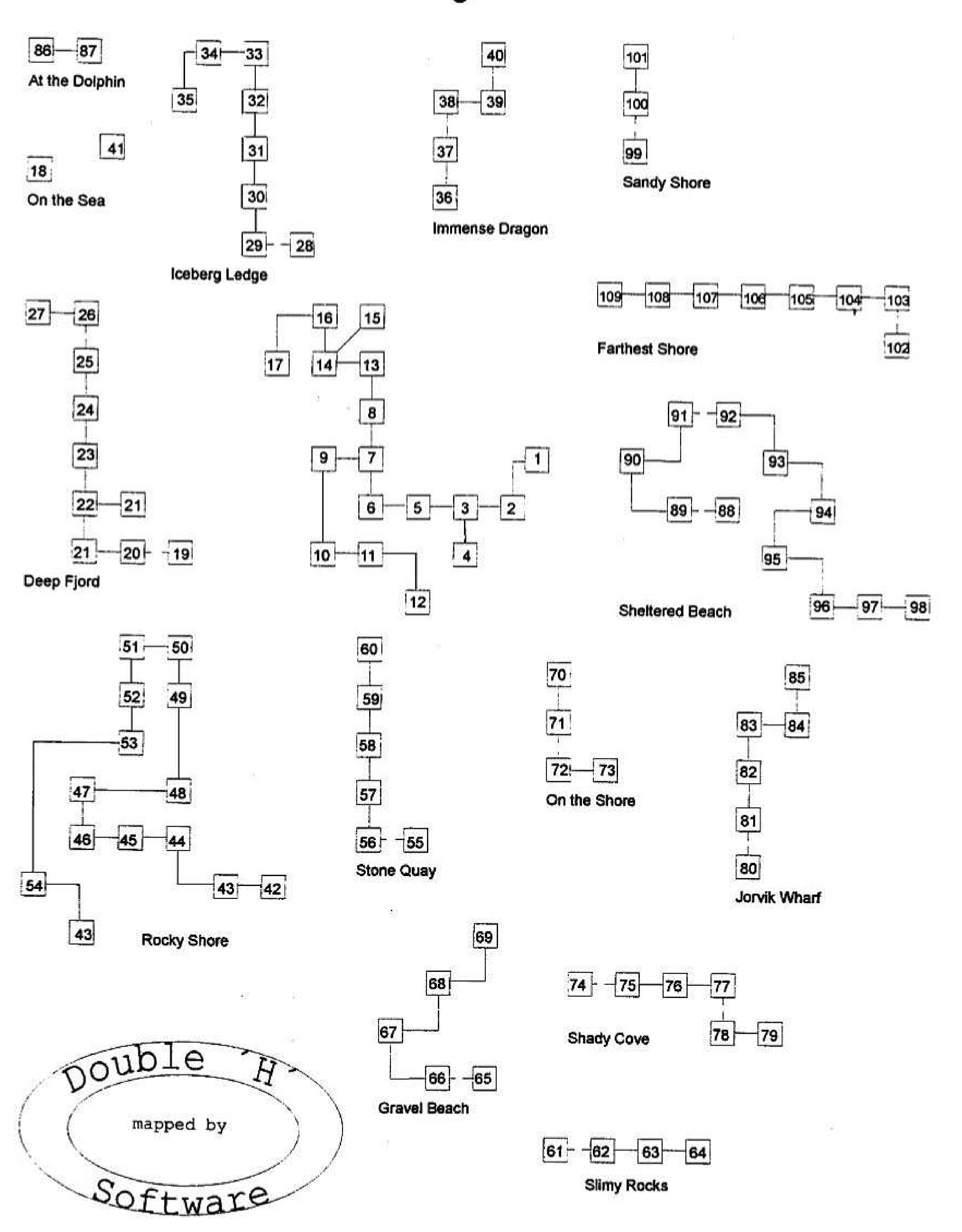
Shore of deep fjord

Out, W, N, N, N, N, In (wir hören ein Gemurmel), W, get chalice (die Stimme fordert uns auf, das Zeug zu trinken - machen wir aber nicht), exam chalice, throw chalice (die böse Zauberin wird vernichtet), look, get mirror, E, Out, S, S, S, E, E, get driftwood, in (wir sind wieder an Bord), alles ablegen, inv, E, S, E, N (wir sollten jetzt am iceberg ledge sein, wenn nicht weiter nach Norden bis wir ihn erreichen).

Iceberg ledge

Get skates, wear skates, get driftwood, get kindling, get whetstone, Out (der Weg nach Norden wird von Eiszapfen versperrt), drop driftwood, light driftwood (das Feuer schmilzt die Eiszapfen und der Weg ist frei), N, N, N, N, W, D, get plug (Wasser dringt ein und spült uns wieder nach oben), E, S, S, S, S, In (wieder an Bord), alles ablegen, inv (wir haben noch die Schlittschuhe an), drop skates, get plug (brauchen wir, um den Maelstrom zu verschliessen). Jetzt gilt es den Drachen zu finden. Wir segeln los....S, S, N und jetzt ca. zwanzig mal nach N und S segeln eventuell auch ein bischen kreuzen. Jedenfalls wenn wir den immensen Drachen treffen geht es so weiter....U, U, E, N, get feather, S, W, D, D (wieder an Bord). Jetzt herumsegeln um an den

Erik The Viking



deep fjord zurückzukehren. Wenn wir dort sind, das Spiel unbedingt speichern. Gleich werdet ihr merken wieso. Jetzt gilt es nämlich den Maelstrom zu finden. Wenn wir also am deep fjord sind geht es in folgende Richtungen. S, S, W. Jetzt müßten wie den Wirbel des Maelstrom sehen. Wenn das nicht der Fall ist, alten Stand einladen und nochmal versuchen. Es kann übrigens mehrere Versuche brauchen, um ans Ziel zu kommen. Wenn wir den Wirbel sehen....throw plug (der Wirbel wird blockiert). Jetzt müssen wir zum rocky shore. Vom Maelstrom aus sind das dann folgende Richtungen....S, W, W, N, N, W, W, W, W, S.

The rocky shore
Out, U, W, W, pull tapestry (wir finden einen Weg nach Norden), N, E, N, N, S, open door, S, open cupboard, look, get bean, eat bean (wir fühlen uns stärker), get scroll, read scroll, drop scroll, get bag, pull lever (wir fallen nach unten), D, In (wieder an Bord), alles ablegen, inv, S, W, S, S, S, S, S, W, S, S, S, S, S, S.

The stone quay
Out, N, N, N, N, open chest (der Spellhound fragt, ob wir sicher sind), yes, look, get jar, get flask, get spittoon, inv, S, S, S, S, In (zurück an Bord), alles ablegen, E, N, N, E, N, N, E, E.

The slimy rocks
Out, E, exam haystack, get needle, E, exam tools (wir finden Schere und Spaten), get clippers, get spade, W, W, In, repair sail (das ist ganz wichtig, da sonst bei unseren weiteren Segelreisen die Segel kaputtgehen), alles ablegen, W, N, E, E.

The gravel beach
Empty pot (wir finden unseren Helm - jetzt aber nicht weiter beachten - das kommt später), get mutton, get spittoon, Out, U, U, U, get stick (der Adler beißt in den Spuknapf), give mutton (der Adler freut sich), get stick, get bracelet, D, D, D, In, alles ablegen, S, W, S, S, E, S (wir sind in der Nähe eines Lochs), S.

The Loch
Get stick, get pot, Out, E, throw stick (es fliegt im Kreis wie ein Bumerang und bringt ein Amulet mit zurück), get amulet, wear amulet (es ist magisch und beinhaltet einen Teleportzauber wenn wir es reiben. Dazu später mehr), fill pot, rub amulet (und schon sind wir zurück auf dem Schiff. Das erspart uns noch so manche Lauferei). Alles ablegen (außer dem Amulet), N, N, E, S, E, E, E, E.

The shady cove
Out, E, E, kneel, pray (eine Tür öffnet sich), In, E, get bell, get candle, get book, read book, drop book, rub amulet. Wenn wir an Bord sind segeln wir weiter...W, W, W, S, S, W, W.

The Jorvik Wharf
Get scales, get bracelet, Out, N, N, E, N, get cat (der Wächter stoppt uns und will zwei Unzen Silber für die Katze), weigh bracelet (wir wiegen das Armband - zufällig genau zwei Unzen), give bracelet, get cat, rub amulet, get jar, tie bell to cat (die Katze schüttelt die Glocke ab), alles ablegen (nicht das Amulet), inv, E, E, S, S, S, S, W, W, N. Jetzt wieder das Spiel saven!! Wir sind jetzt am sheltered beach und müssen nunmehr den Delphin finden. Deshalb segeln wir auf der Route in Richtung Jorvik Wharf. Hier müßte uns der Delphin begegnen. S, E, E, N, N, N, N, W, W. Auf dieser Route muß er sein. Wenn wir ihn nicht angetroffen haben, laden wir den Spielstand wieder ein und versuchen es erneut. Das kann mehrmals so gehen und dauert oft, bis man Glück hat. Wenn wir den Delphin treffen..get flask, Out, fill flask, IN. Jetzt segeln wir zurück zum sheltered beach.

The sheltered beach
Inv (wir haben noch die Flasche mit Luft und das Amulet). Drop flask, get whetstone, get candle, light candle, drop whetstone, get helmet, wear helmet, get hammer, get clippers, Out, U, U, hit slab (sie öffnet sich), In, D, D, D, D, E, E (wir finden 7 schnarchende Zwerge vor), exam dwarfs (es sind Frauen mit Bärten - Monty Python läßt grüßen), get beard, get beard, rub amulet (wieder an Bord), alles ablegen, Inv, drop helmet, S, W, W, N, N, N, NW, N, N, N, N, N, W, N, W.

The sandy shore
Get pot, get spade, Out, N, exam tree (voller Steinfrüchte), shake tree (eine Frucht fällt herab), look, get fruit, plant fruit, empty pot (Wasser läuft aus und die Frucht beginnt zu wachsen), look, get roots, rub amulet, alles ablegen, Inv (wir haben nur noch das Amulet), E,S, E,S, S, S, S, S, S.

The stone quay (2. Besuch)
Get jar, get flask, Inv (nachsehen ob auch Luft drin ist), get spittoon, get sinew, get roots, get beard, Out, N, N, N, N (Kasarmi fragt uns, ob wir alle Zutaten haben), yes (er nimmt alles und etwas erscheint), look (wir sehen ein gelbes Band), get ribbon, rub amulet. Alles ablegen, Inv, E, N, N, W, N.

The farthest shore
Get ribbon, get mirror, Out, W, W, W, W, W, read inscription, tie ribbon, drop mirror, drop amulet, look (wir sehen den Wolf angebunden), open door, W......The END
Erik hat seine Familie wieder gefunden. Das Adventure ist gelöst. Wir hoffen, ihr hattet wieder Spaß an der Sache. Bis demnächst hier an gleicher Stelle.....

Harald R. Lack, Heidenauer Str. 5, 83064 Raubling
Hub. Kracher, Schulweg 6, 83064 Großholzhausen

ANTICOPY



by Miles Kinloch

I thought I'd send you another little routine I wrote to demonstrate a hypothetical application for the 'hidden tracks' idea. It's probably of little practical use these days, given the decline in demand for commercial Spectrumsoftware, but I thought the idea itself may be of interest to people as it illustrates a possible technique. The secret of its success would have lain in the success of its secrecy, as the key to its effectiveness would have been unawareness of the mechanism rather than any inherent technical obstacles to be overcome.

As the name suggests, ANTICOPY prevents a working copy of program being made on any medium other than the original disk on which it is supplied. It does this by testing for the presence of track 80, and then carries out a further check on the first 20 bytes of its first sector, which must add up to a specified checksum. (Only the low-byte is tested, i.e. the checksum valueis modulo 256.) To allow the tracks to be used simultaneously for other purposes, the value can be altered to suit existing data.

If the copy is run on a system without a +D or Disciple attached, the first hurdle of Command Code 63 will cause it to fail. Should it get this far but find track 80 does not exist or contains a wrong checksum, the routine will then print a warning message and reset the system on the next keypress.

To disguise their appearance, the characters in this message all have BIT 7 set. Further anti-hacking measures include calls to the Spectrum ROM via RST 16 DW nnnnn, which are unlikely to be displayed as such by most disassemblers, while the shadow RAM is also reset to prevent a snapshot being made at this stage. If the routine were used seriously, the code could in addition be saved to auto-run, thus eliminating the tell-tale USR all... but as it's only an academic exercise, I'll avoid getting too carried away!

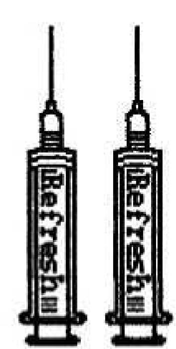
```
  10 REM ANTICOPY DISK FORMATTER
  20 CLS: INPUT "Which drive? ";d
  30 POKE @d,210: FORMAT dd: POKE @d,
     208: REM Create new tracks
  40 POKE 16384,100: SAVE @d,80,1,
     16384: REM Sector 1 checksum
  50 STOP
9999 SAVE d*"FORMATTER" LINE 10
```

```
  10 REM CLONE-PROOF PROTECTION
  20 REM
  30 REM AN EXAMPLE APPLICATION
  40 REM FOR HIDDEN TRACKS IDEA
  50 REM
  60 REM By Miles Kinloch, 1997
  70 REM
  80 BORDER 1: CLEAR 59999: GO SUB 120
  90 PRINT "ANTICOPY ROUTINE"''"(113
     BYTES RELOCATABLE CODE,"'"CURRENT
     LY AT ADRESS 60000.)"''"ACTION:"''
     "PRINTS THE MESSAGE 'THIS COPY IS
     NOT AUTHORISED' AND RESETS THE SP
     ECTRUM AND +D. WORKS WITH G-DOS
     AND BETA-DOS, AND SHOULD ALSO BE
     OK ON UNIDOS & DISCIPLE (THOUGH
     NOT TESTED ON THESE)."''"TEST CRI
     TERIA: THE ORIGINAL DISK MUST BE
     OF THE STANDARD, DOUBLE-SIDED DO
     UBLE-DENSITY TYPE, FORMATTED WITH
     A TRACKS POKE (@1/@2) OF 210. THE
     FIRST 20 BYTES IN SECTOR 80:1 MU
     ST ALSO ADD UP TO A CHECKSUM OF
     100. THIS VALUE IS IN START+24 &
     CAN BE CHANGED."
 100 PRINT #0;AT 0,0;"PRESS 'T' TO TEST
     ROUTINE, ANY OTHER KEY TO SAVE MA
     CHINE CODE.": PAUSE 0: IF INKEY$=
     "T" OR INKEY$="t" THEN RANDOMIZE
     USR 60000: CLS: PRINT"    TEST PAS
     SED - GENUINE COPY": STOP
 110 SAVE d*"Anticopy.c"CODE 60000,113:
     STOP
 120 REM Code Poker
 130 RESTORE: LET c=0: FOR a=60000 TO
     60012: READ b: POKE a,b: LET c=c+
     b*(a<>60024): NEXT a: IF c<>14721
     THEN PRINT "ERROR IN DATA!": STOP
 140 RETURN
 150 DATA 17,1,80,207,63,118,205,82,0,
     59,59,209
 160 DATA 56,12,1,0,20,207,60,129,79,
     16,250,254
 165 REM
 170 DATA 100: REM Checksum byte - may
     be changed if desired
 175 REM
 180 DATA 200,243,213,207,71,215,107,
     13,62,2,215
 190 DATA 1,22,225,17,71,0,25,126,254,
     255,40,12
 200 DATA 254,32,40,2,230,95,215,242,
     21,35,24,239
 210 DATA 33,255,63,54,0,43,124,183,32,
     249,175,219
 220 DATA 254,31,56,250,205,80,0,199,
     32,32,244,232
 230 DATA 233,243,32,227,239,240,249,
     32,233,243,32,238
 240 DATA 239,244,32,225,245,244,232,
     239,242,233,243,229
 250 DATA 228,32,32,32,255
9998 REM
9999 SAVE d*"ANTICOPY" LINE 10
```

"ANTICOPY" COMMENTED ASSEMBLY

```
60000 017 001 080    LD DE,20481       ; DE=Track/Sector (i.e. 80:1).
60003 207 063        RST 008 DB 063    ; Read sector at DE. (C flag set afterwards
                                       ; if error.)
60005 118            HALT              ; Wait first for an interrupt...
60006 205 082 000    CALL 00082        ; ...then call RET in Spectrum ROM...
60009 059            DEC SP            ; ...in order to interrogate stack and...
60010 059            DEC SP            ; ...find program counter (to calculate
                                       ; message address).
60011 209            POP DE            ; (Program counter POPped into DE.)
60012 056 012        JR C,60026        ; C flag after RST 8 DB 63 (set if track
                                       ; non-existent).
60014 001 000 020    LD BC,05120       ; B=counter for DJNZ loop, C=checksum
                                       ; counter.
60017 207 060        RST 008 DB 060    ; Get each byte of loaded sector into A...
60019 129            ADD A,C           ; ...add to checksum counter...
60020 079            LD C,A            ; ...transfer result to C register...
60021 016 250        DJNZ 60017        ; ...and loop for 20 bytes.
60023 254 100        CP 100            ; Compare final result with value expected.
60025 200            RET Z             ; Return if this tallies, i.e. disk is bona
                                       ; fide.
60026 243            DI                ; Come here if checksum wrong or track
                                       ; doesn't exist.
60027 213            PUSH DE           ; Push program counter value (still in DE).
60028 207 071        RST 008 DB 071    ; Page in +D/Disciple.
60030 215 107 013    RST 016 DW 03435  ; Call CLS in Spectrum ROM.
60033 062 002        LD A,002          ; Make stream 2 current (for printing to
                                       ; screen)...
60035 215 001 022    RST 016 DW 05633  ; ...and call 'channel open' routine in
                                       ; Spectrum ROM.
60038 225            POP HL            ; Original program counter value now in HL.
60039 017 071 000    LD DE,00071       ; Add offset of 71...
60042 025            ADD HL,DE         ; ...to calculate start address of text
                                       ; message.
60043 126            LD A,(HL)         ; Get each character into A.
60044 254 255        CP 255            ; Test for end of message.
60046 040 012        JR Z,60060        ; Move to next stage (reset) if finished
                                       ; printing.
60048 254 032        CP 032            ; (Spaces excepted from BIT 7 encoding.)
60050 040 002        JR Z,60054        ;
60052 230 095        AND 095           ; Reset BIT 7 (and BIT 5 to make capital
                                       ; letters) to obtain true ASCII character.
60054 215 242 021    RST 016 DW 05618  ; Call print routine in Spectrum ROM.
60057 035            INC HL            ; Increment address for next character to
                                       ; be printed...
60058 024 239        JR 60043          ; ...and loop back (until end-marker is
                                       ; reached).
60060 033 255 063    LD HL,16383       ; Point HL to top of shadow RAM.
60063 054 000        LD (HL),000       ; Reset each byte in turn to zero...until
60065 043            DEC HL            ; ...all addresses done (except first 256).
60066 124            LD A,H            ; (8192-16383=+D RAM, 0-8191=Disciple RAM.)
60067 183            OR A              ;
60068 032 249        JR NZ,60063       ;
60070 175            XOR A             ; Wait for keypress...
60071 219 254        IN A,(254)        ;
60073 031            RRA               ;
60074 056 250        JR C,60070        ;
60076 205 080 000    CALL 00080        ; Page out +D/Disciple.
60079 199            RST 000           ; Reset Spectrum.
(60080-60111)        DM "  THIS COPY IS NOT AUTHORISED   " ; (Disguised with BIT
                                       ; 7 set.)
60112 255            DB 255            ; End-of-text marker
```

Die grauen Zellen des Spectrum



Teil 4: Refresh

Der "Haupt"-Speicher des Spectrum besteht aus dynamischen RAMS (DRAMS). Bei diesen muß im Gegensatz zu statischem RAM der Speicherinhalt regelmäßig "aufgefrischt" werden. Geschieht dies nicht, verliert die Speicherzelle nach einer gewissen Zeit ihre Information. Man könnte nun denken, klar, das sehen wir beim Anschließen des Spectrums an das Netzteil: der vorher vorhandene Screeninhalt ist weg, weil die Elektronik, die die Speicherinhalte auffrischt (sog. Refresh) keinen Strom hatte. Falsch! Das gleiche würde auch bei statischen RAMs passieren, auch diese verlieren ohne Batteriepufferung ihren Inhalt, wenn die Versorgungsspannung ausfällt.
Zum Erhalten der Speicherinformationen genügt es, mindestens alle 2 Millisekunden jede Zeile der Speicherzellenmatrix einmal zu adressieren, also 500 mal pro Sekunde. Dies geschieht sogar ohne weitere Vorkehrungen, denn 50 mal pro Sekunde muß auf den Bildspeicher zugegriffen werden, um das Bild anzuzeigen! Genauer gesagt sind es 50 Halbbilder pro Sekunde = 25 Vollbilder. Da ein 16 KBit-RAM-Chip 128 Zeilen und Spalten hat und jeder Chip 1 Bit eines Bytes ausgibt, wird schon bei 128 Bytes, die gelesen werden, jede Zeile einmal adressiert. 32 Bytes sind eine Screen-Zeile, 128 Bytes also 4 Zeilen. Das sind 4/192 = 1/48 des gesamten Bildes. Demnach wiederholt sich das Adressieren der 128 Zeilen 48 mal pro Bild, also 48 * 25 = 1200 mal pro Sekunde. Hurrah, das ist öfter als die geforderten 500 mal pro Sekunde. Beim Rücklauf des Elektronenstrahls in der vertikalen Synchronisationsphase entsteht jedoch eine Pause von 5 Millisekunden, während derer nicht auf den Speicher zugegriffen wird. In dieser Zeit muß die CPU den Refresh durchführen. Dazu besitzt der Z80 Prozessor einen Ausgang RFSH (Refresh). Ist dieser Low, hat der Prozessor eine 7 bit-Adresse zur Speicherauffrischung auf den Adreßbus gelegt. Im Gegensatz zu Lese- und Schreiboperationen wird jedoch weder RD noch WR (Lese- bzw. Schreibvorgang) von der CPU aktiviert. Der RFSH-Ausgang geht zur ULA, die die Adresse dann an die DRAMS weiterschaltet, indem sie das RAS-Signal an die DRAMS legt.
Wie in Teil 2 der Serie beschrieben, werden bei einer 32 KB Speichererweiterung die Signale RAS und CAS nicht von der ULA erzeugt, sondern von Logikgattern. Diese aktivieren die RAS-Leitung der Speicher, sobald MREQ an der CPU auf LOW gesetzt wird. Auch dies ist nur während der Video-Synchronisation nötig, denn die unteren 7 Bit des Adreßbus werden während der normalen Videobild-Erzeugung ständig gelesen und diese unteren 7 Bit stellen die Zeilenadresse dar.
Für diejenigen, die ausprobieren wollen, wie ohne Refresh Inhalte verloren gehen: Wenn der Pin 28 (RFSH) des Z80 nicht mehr mit der ULA direkt verbunden ist (Leiterbahn durchtrennen), geht der Bildspeicher und das übrige RAM während der Videosynchronisation "flöten".
So, das ist das Ende unserer kleinen Serie. Ich hoffe, es hat Spaß gemact.

Andreas Schönborn, Gössingstraße 44
44319 Dortmund, Tel. 0231/217103
e-mail: schoenbo@hottinger.de

Multi(t)asking

eine Antwort an J. Koelman (Info 95, Seite 13)

Als die Begriffe neu waren, gab es beide: Multiasking (Fragen) und Multitasking (Aufgaben). Da es in meinem Beitrag um die Aufgabe der mehrfachen Abfrage ging, habe ich das (t) in Klammern gesetzt. Es handelte sich also hierbei um eine Art Wortspielerei.
Der sehr interessante Artikel von Herrn Koelman ist meines Erachtens auch kein echtes Multitasking, sondern eine schnelle Umschaltmöglichkeit dreier Programme. Eines ist in Betrieb, während zwei ruhen. Multitasking sollte wohl sein, daß eine Aufgabe mit dem Anwender zusammen bearbeitet wird, während eine andere oder mehrere währendessen vom Computer alleine abgehandelt wird.

Herbert Hartig, Buchloe

Ich möchte hier etwas Klärendes zum Begriff "Multitasking" beitragen: Beim Multitasking werden einzelnen Programmen Zeitzyklen zugeteilt, in denen jeweils ein oder mehrere Programmschritte abgearbeitet werden, bevor das nächste Programm aufgerufen wird. Die Programme werden also tatsächlich abwechselnd abgearbeitet. Ohne eine übergeordnete Software, die diese Zeitabläufe steuert, ist dies jedoch nicht möglich. Ein Beispiel aus meinem Arbeitsbereich für Multitasking ist z.B. das Arbeiten in einem Programm währenddessen Daten zum Ausdruck an einen Drucker gesendet und gleichzeitig Daten per ISDN empfangen werden.

Wo vom WoMo-Team

Ja, die gute Weihnachtszeit mit ihren für viele freien Tagen. Sie bescherte uns und euch dank einer großartigen Fleißarbeit von Nele Abels-Ludwig diesmal ein

Thematisches Jahresinhaltsverzeichnis für 1997

(Teil 1)

Die Einträge erfolgen innerhalb der Sachgruppen chronologisch, wobei thematische Überschneidungen leider nicht immer zu vermeiden sind, aber sooo lang ist das Verzeichnis ja auch nicht! Das Format ist:

Monat (Seite), "Titel", Autor: Kurzinhalt

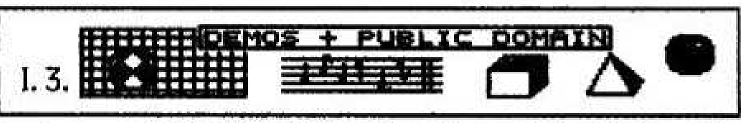
Nicht aufgenommen sind die meisten redaktionellen Beiträge. Artikel der Rubrik "Die Seite für den SAM!" sind den entsprechenden Sachgruppen untergeordnet und mit (SAM) gekennzeichnet.

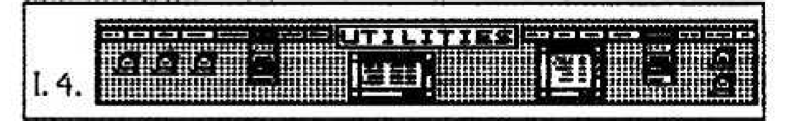
I. SOFTWARE

I.1. SPIELE

I. 2. TEXTADVENTURES

I. 3.





• Januar (9), "Pick-Poke-It", Heinz Schober: Beschreibung eines +D Hack-Utilites von Drag Soft
• August (8), "Datei mit 22,5-facher Kapazität", Herbert Hartig: Herbert hat sein Datenbank-programm auf OPUS konvertiert
• November (14), "Multistore", Miles Kinloch: PD-Utility für das Multiface
• November (15), "Tornado", Roelof Koning: Bedienung des Editor/Assemblers (+D)
• Dezember (4), "Outwrite v2D", Ingo Wesenack: Kurzbeschreibung des Texteditors (SAM)

I.5. SONSTIGES

Februar (3), "Software-Test: Syncytium", WoMo: Software-Sampler mit Spielen und Utilities (SAM)
• Februar (14), "DTP leicht gemacht", Wo von WoMo: DTP mit dem PCG Spectrum Desk Top Publishing Package
• März (7), "DTP leicht gemacht (Teil 2), Wo von WoMo: s. o.
• Oktober (8), "SAM-Software von Persona", WoMo: Software für den (SAM)

II. PROGRAMMIEREN

• Januar (11), "PCX-Grafik auf dem Spectrum", Helge Keller: Utility zum Konvertieren von schwarzweißer PCX-Grafiken
• Januar (12), "In der Kürze liegt die Würze (Teil 2)", Rupert Hoffmann: Fernschreiberartige Text-ausgabe auf dem Bildschirm
• Februar (10), "BASIC für Anfänger (Teil 5)", P. C. R.: Basic-Kurs
• März (4), "How to fill a block of memory with a value?", Dalnikovas Eugenijus, Blockfüllen in Maschinensprache
• März (6), "In der Kürze liegt die Würze (Teil 3), Rupert Hoffmann: Mehrfarbiger Bildschirmrand unter BASIC
• März (6), "Der Spectrum kann lesen", Herbert Hartig: Verwendung von SCREEN$
• März (8), "BASIC für Anfänger (Teil 6)", P. C. R.: Basic-Kurs
• April (6), "Programmier Wettbewerb: and the winner is...", Peter Meindl: Auswertung des Laubsägewettbewerbs von 1996
• April (9), "Drehrad", Nele Abels-Ludwig: Drehrad zur Arbeitsanzeige
• Mai (5), "Fraktale Grafiken für Basic-Anfänger (Teil 1)", Nele Abels-Ludwig: Apfelmännchen-Programmierung in Basic
• Mai (11), "Headerless laden", Miles Kinloch: headerlose Nachladeteile vom Tape laden
• Mai (14), "Programmanalyse von Druckroutinen in Basic", Herbert Hartig: Druckerpatches für z. B. View Royal
• Juni (2), "Zusätzliche Funktionen des 128k-Editors", Nele Abels-Ludwig: Editierfunktionen im 128k-Basic
• Juni (7), "VU-3D für den Kempston Joy", Miles Kinloch: Patch zur Anpassung an den Kempston Joystick
• Juni (7), "Autostop", WoMo: Programme so laden, daß sie nicht automatisch starten
• Juni (8), "Fraktale Grafiken für Basic Anfänger (Teil 2)", Nele Abels-Ludwig: Apfelmännchen-Programmierung in Basic
• Juni (15), "Coding aus Rußland", FBI of the Unbelievables: Anmerkungen zu März (4), "How to fill a block..."
• Juni (15), "Terminkalender, Gedenktage, Ereignisse", Herbert Hartig: für Vergeßliche
• Juli (5), "In der Kürze liegt die Würze (Teil 4)", Rupert Hoffmann: Rosettengrafik und Verwendung von Point
• Juli (8), "Fraktale Grafiken für Basic-Anfänger (Teil 3)", Nele Abels-Ludwig: Apfelmännchen-Programmierung in Basic

Fortsetzung im nächsten Info

NEWSFLASH

Andy Davis von AlchNews hat jetzt Teile seiner PD-Bibliothek im Netz. Sie können als gezipte Plus D Diskimages geladen werden, Transfer-software für den PC liegt bei. Bis jetzt liegen ein rundes Dutzend Images mit jeweils 50-60 Programmen vor:

http://www.cix.co.uk/~mharrop/index.htm

Siehe auch die Seiten von AlchNews:

http://www.alchemist.clara.net/

HOBBYAUFLØSUNG

2 ZX Spectrum 48K (Gummitastatur), 1 Betadisk-Interface + 2 Laufwerke (5,25 Zoll, 2 x 40 bzw. 2 x 80 Spuren) plus Netzteil, Kassettenrecorder mit Bandzählwerk, S/W-Kofferfernseher als Monitor, 9-Nadel-Drucker mit Interface, diverse Baugruppen (Joystick, Sound), viele Programme auf ca. 60 Disketten (Spiele, Programmierung, Anwendung) - teilweise auch Kassetten, Literatur (Hardware, Programmierung, Handbücher), diverse Ersatzteile. Alles zusammen für ca. 550,- DM an Selbstabholer.

Jens Mückenheim, Heidelbergstraße 20 06577 Braunsroda, telefonische Absprache unter: 0171/487 0417 (ab 18 Uhr)